

ABSTRACT

A mechanism is described for transient versioning in architectures that manage node ranges, wherein each node is assigned a node ID value and a set of nodes form a range of node IDs called a node range. Each entry in the index describes one range and points to where the range is located. Individual nodes are located by finding the correct range in the index. When nodes are added to or deleted from a node range, the range of nodes are versioned by copying the nodes before changes, to transient storage, and then the original nodes are modified. Different versions are tracked by assigning timestamps to each copy of the node range. Each entry in the node ID range index points to the location of the nodes in a range called the range identifier or RID. Before changes are made in a range, the nodes in a range are copied to a Version Hash Table based on the RID. Copies of the range including the current one is assigned a timestamp or LSN. New readers after a change, access the current nodes through RID, while old readers access the old nodes through the same RID, but hashing it to find the shadowed copy in the Version Hash Table. If changes causes nodes in the range to be moved to a new RID, previous readers need to be redirected from the new RID to the old RID.

